Master's Thesis

Learning What to Remember: Long-Term Episodic Memory Network For Learning From Streaming Data

Moonsu Han

Department of Computer Science and Engineering

Graduate School of UNIST

2019

Learning What to Remember: Long-Term Episodic Memory Network For Learning From Streaming Data

Moonsu Han

Department of Computer Science and Engineering

Graduate School of UNIST

Learning What to Remember: Long-Term Episodic Memory Network For Learning From Streaming Data

A thesis/dissertation
submitted to the Graduate School of UNIST
in partial fulfillment of the
requirements for the degree of
Master of Science

Moonsu Han

12. 11. 2018.

Approved by

Advisor
Tsz-Chiu Au

ULSAN NATIONAL INSTITUTE OF SCIENCE AND TECHNOLOGY

Learning What to Remember: Long-Term Episodic Memory Network For Learning From Streaming Data

Moonsu Han

This certifies that the thesis/dissertation of Moonsu Han is approved.

12. 11. 2018.

Advisor: Tsz-Chiu Au

Sungju Hwang

Jaesik Choi

Abstract

Current generation of memory-augmented neural networks has limited scalability as they cannot efficiently process data that are too large to fit in the external memory storage. One example of this is lifelong learning scenario where the model receives unlimited length of data stream as an input which contains vast majority of uninformative entries. We tackle this problem by proposing a memory network fit for long-term lifelong learning scenario, which we refer to as Long-term Episodic Memory Networks (LEMN) that features a RNN-based retention agent that learns to replace less important memory entries based on the retention probability generated on each entry that is learned to identify data instances of generic importance relative to other memory entries, as well as its historical importance. Such learning of retention agent allows our long-term episodic memory network to retain memory entries of generic importance for a given task. W validate our model on a path-finding task as well as synthetic and real question answering tasks, on which our model achieves significant improvements over the memory augmented networks with rule-based memory scheduling as well as an RL-based baseline that doesn't consider relative or historical importance of the memory.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The tremendous success of deep learning has led to many changes in natural language processing. The basis of this success is that it can be possible to automate the feature extraction using the gradient descent algorithm in the optimization theory and can express the data more variously than conventional linear method using nonlinearity. Also, the advent of high performance graphic card allows us to extract high-level representation from the layers to be deep. Unlike image classification and object detection, in natural language processing, it is important to understand the relationship between words in a sentence and to deduce implicit information. Because the sentence is organized by words, and words are organized by letters, they can be viewed as time-series data. A method for analyzing time-series data based on deep learning is typically Recurrent Neural Network. The most famous RNN techniques, Long Short-Term Memory (LSTM, Hochreiter & Schmidhuber, 1997) and Gated Recurrent Unit (GRU, Chung et al., 2014) have shown outstanding performance in translation, generating a sentence, video analysis, and question-answering. In order to extract important information from time-series data, RNN defines a fixed dimensional vector to memorize meaningful information. While continuing to read new information at each time, it controls the degree of retention of previous information and the degree of storage of new information using the activation function. However, since RNN uses memory as a fixed dimensional vector, it has limitations in long-term memory capacity. Also, since all the weights for learning should be stored in the graphic card, the deep learning model can only learn data in time-series within a limited capacity. In other words, RNN is efficient in analyzing limited time information within limited capacity, but RNN is vulnerable if it has to maintain long time information to achieve the task.

There have been many studies to overcome the weaknesses of long-term memory such as Neural Turing machine (NTM, Graves et al., 2014) and End-to-End Memory Networks (Sukhbaatar et al., 2015). They used read/write mechanism to memorize the information that could be used to solve the task and the attention mechanism to ensure the interpretability between the representation of input which is called a memory cell and the representation of output. They proved that their networks can only remember the information needed to solve the task. Also, they used the method to control the address of the memory cells using a content-based access or the index of memory cell. They experimented on memory-oriented tasks such as copying, pathfinding, and question answering. However, their research was conducted on short-term and toy task data rather than long-term and complex data, and they controlled only a small amount of memory, rather than controlling large amounts of memory. Therefore, it is necessary to study memory handling in a large amount of complex data.

To create neural network architecture which can process large amounts of streaming data and manage the memory cells such that the external memory module, we have to consider efficient management method while reading the streaming data. Namely, our architecture can recognize the information that is used to solve the task and can store them in the memory cells. It should be able to replace useless information in the memory cells with new information which will be important in the future. Also, access to large data is required, which is not possible with current methods due to the limitations of graphic card capacity. We try to solve these problems using reinforcement learning to train the neural network and to manage the memory cells while maintaining useful information from the streaming data.

How can we then select which memory entries are most important? To achieve this goal, we propose to train the memory module itself using reinforcement learning to delete the most uninformative memory entry in order to maximize its reward on a future task. However, this is a challenging task since for most of the time, this scheduling should be performed without knowing which task will arrive when. Thus, deciding which memory to keep and which to erase should be done by considering relative generic importance of the memory entries. To tackle this challenge, we implemented the memory retention mechanism using a spatio-temporal recurrent neural network, that can learn relative importance among the memory entries as well as their historic importance. We refer to this memory-augmented networks with spatio-temporal retention mechanism as Long Term Episodic Memory Networks (LEMN). LEMN can perform selective memorization to keep a compact set of important pieces of data that will be useful for future tasks in lifelong learning scenarios. We validate our proposed retention mechanism against naive scheduling method as well as RL-based scheduling on three different tasks, namely path-finding, episodic question answering and long question answering, against which it significantly outperforms.

Our contribution is twofold:

1. We consider a novel task of learning from streaming data, where the size of the memory is significantly smaller than the length of the data stream.
2. To implement a retention mechanism, we propose a retention agent that can be integrated with existing external memory neural networks, which is trained with reinforcement learning to keep memory cells of general importance.

Chapter 2

Related Work

2.1 Memory-Augmented Networks

Neural Turing Machine (NTM, Graves et al., 2014) proposes the concept of memory and the controller that reads from and writes to memory. The controller composed of read heads and write heads uses soft-attention mechanism to access memory for differentiable memory access. The NTM has two addressing mechanisms: content-based addressing and location based addressing. Location-based addressing allows a data sequence to be stored in consecutive memory cells to preserve its sequential order. However, once the write head accesses a distant memory cells, the sequential order of information in consecutive memory cells is no longer preserved. (Graves et al., 2016) propose Differentiable Neural Computer that extends the NTM to address the issue by introducing a temporal link matrix. As it is costly to write into memory while preserving its sequential order, memory-augmented networks with write mechanism are mostly used in cases where it is not necessary to track which sequential order the memory has been written (Santoro et al., 2016; Vinyals et al., 2016; Kaiser et al., 2017; Kim et al., 2018). Unlike NTM, End-to-End Memory Networks (MemN2N) (Sukhbaatar et al., 2015) and Dynamic Memory Networks (DMN+) (Xiong et al., 2016) don't have write mechanism but store all the inputs into memory. For this reason, the sequential order of the inputs are preserved at no extra cost; thus they are more suitable for episodic question answering problems such as bAbI tasks (Weston et al., 2015). In addition, they have sophisticated read mechanisms that allow to reason through multiple hops (or passes in DMN+). (Seo et al., 2016) propose more advanced soft-attention based read mechanism, Bi-Directional Attention Flow (BiDAF), which obtains impressive performance on a difficult reading comprehension dataset, Stanford Question Answering Dataset (SQuAD, Rajpurkar et al., 2016). (Oh et al., 2016) extend MemN2N to train a deep reinforcement learning agent, Feedback Recurrent Memory Q-Network (FRMQN) to solve 3D Mazes. However, such neural networks have the limitation that the size of external memory should be large enough to store all the data.

2.2 Memory Retention Policy

Most of the existing approaches (MemN2N, DMN+, BiDAF and FRMQN) don't consider the case where memory becomes full, and simply truncate the sequence of data to the size of memory if it is longer than the memory size. Many of mutable external memory neural networks employ least-recently-used (LRU) based memory retention policy, which overwrites new data into the least used memory cell. This policy, although more reasonable than FIFO, is still limited as it is a hand-crafted rule and doesn't consider actual long-term dependencies between a memory entry and the task. This is a critical limitation in lifelong learning setting, since some of the memory entries written in the long past and have not been accessed for long may still be necessary to respond to queries that arrive in the far future. Our work, on the other hand, is able to learn such long-term dependencies. DNTM (Gulcehre et al., 2016) learns where to overwrite using reinforcement learning as done in our work.

Yet, it only considers the pairwise relationships between the current data instance and each individual memory entries, while our model learns both relative importance and historic importance of a memory entry using a spatio-temporal RNN architecture.

Chapter 3

Approach

3.1 Learning What To Remember From Streaming Data

We consider the problem of learning from a long data stream that contains large portion of unimportant, noisy data (e.g. routine greetings in dialogs) with limited memory. Formally, an agent A takes as input a streaming data (e.g. sentences or images) $X = \{x_i, \ldots, x_T\}$ and manages an external memory $M = \{m_i, \ldots, x_N\}, m_i \in R^d$ where $T \gg N$. Thus, the agent should decide which memory cell to evict to store incoming data. To this end, the agent should learn the relative importance of memory entries for a future task. In traditional reinforcement learning scheme, we can formulate this problem as learning the policy $\pi(\alpha_t|s_t)$ to maximize a return R, where action a_t is a memory cell to erase, and state $s_t = [M_t; x_t]$, where M_t is the current memory and x_t is the input at time t. The agent appends x_t to the memory M_t until it reaches the maximum size N. If the memory is full, the agent erases one memory cell based on the policy $\pi(m_i|M_t, x_t)$ to append x_t. Thus, we can consider $1 - \pi(m_i|M_t, x_t)$ as the retention value of each memory. At arbitrary time step t_f, it encounters a task T (e.g. question answering) with a reward R_T, whose reward is defined by the specific task. For QA task, the reward will be +1 if it generated a correct answer and −1 otherwise. The instance x_t which arrives at timestep t can be in any data format, and is transformed into a memory vector representation $e_t \in R^d$ to be stored in memory.

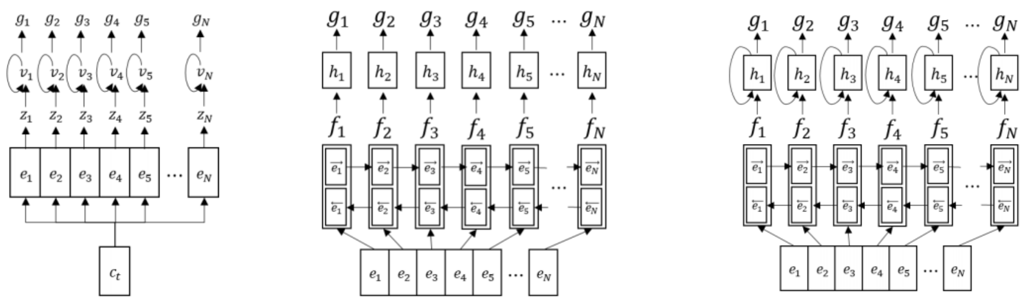


Fig 1. Illustration of methods: *Input-Matching LEMN (IM-LEMN) (Left), Spatial LESM (S-LEMN) (Center), Spatio-Temporal LEMN (ST-LEMN) (Right).*

3.1.1 Memory-Retention Mechanism

In this section, we describe three different memory-retention mechanisms in detail. We first encode input x_t and each memory cell $m_{t,i}$ to a vector representation as follows:

$$c_t = \psi(x_t) \tag{1}$$

$$e_{t,i} = \phi(m_{t,i}) \tag{2}$$

where c_t, $e_{t,i} \in R^d$, $\psi(\cdot)$ can be an input embedding similar to RNN controller in (Gulcehre et al., 2016) or position encoding in (Sukhbaatar et al., 2015) and $\phi(\cdot)$ can be a memory encoding layer of the base external memory neural network. For example, MemN2N (Sukhbaatar et al., 2015) uses a bag of words and embedding matrices to convert to memory representation.

3.1.2 Input-Matching LEMN

This is the simplest RL-based retention mechanism that is similar to DNTM from (Gulcehre et al., 2016) and utilizes the memory usage information, which we refer to as Input-Matching LEMN (IM-LEMN). The usage of each memory representation mi is computed as the learned similarity between it and the current input x_t. As in (Gulcehre et al., 2016), we least recently used (LRU) addressing by computing the exponential moving average v_t of the logit z_t, the LRU factor γ_t, and the policy as follows:

$$v_{t,i} = 0.1 v_{t-1,i} + 0.9 z_{t,i} \tag{3}$$

$$\gamma_t = \sigma(W_\gamma v_t + b_\gamma) \tag{4}$$

$$g_{t,i} = z_{t,i} - \gamma_t v_{t-1,i} \tag{5}$$

$$\pi(m_i | M_t, x_t) = softmax(g_{t,i}) \tag{6}$$

where $W_\gamma \in R^d$, $b_\gamma \in R^d$, $softmax(z_i)$ is $exp(z_j)$, $\sigma(z)$ is $1/(1 + exp(z))$. This model estimates the policy based on the similarities between input x_t and memory cells $m_{t,i}$.

3.1.3 Spatial LEMN

A major drawback of IM-LEMN is that the score of each memory depends only on the input x_t. In other words, the score is computed independently between memory cell and input but doesn't consider its relative importance to other data instances in the memory. To overcome this limitation, we propose Spatial LEMN (S-LEMN) which computes the relative importance of memory cells to its neighbors and other memory cells using a bidirectional GRU as follows:

$$\vec{e}_{t,i} = GRU_{\theta_{fw}}\left(e_{t,i}, \vec{e}_{t,i-1}\right) \tag{7}$$

$$\overleftarrow{e}_{t,i} = GRU_{\theta_{bw}}\left(e_{t,i}, \overleftarrow{e}_{t,i+1}\right) \tag{8}$$

$$f_{t,i} = ReLU\left(W_f\left[\vec{e}_{t,i}, \overleftarrow{e}_{t,i}\right] + b_f\right) \tag{9}$$

where $W_f \in R^{2d\times d}$, $b_f \in R^d$, GRU_θ is a Gated Recurrent Unit parameterized by θ, $[\vec{e}_{t,i}, \overleftarrow{e}_{t,i}]$ is a concatenation of features, ReLU is a rectified linear unit. We use multi-layer perceptron (MLP) with scalar output to estimate the policy as follows:

$$h_{t,i} = W_h f_{t,i} + b_h \tag{10}$$

$$g_{t,i} = W_g h_{t,i} + b_g \tag{11}$$

$$\pi(m_i|M_t, x_t) = softmax\left(g_{t,i}\right) \tag{12}$$

where $W_h \in R^{d\times d/4}$, $b_h \in R^{d/4}$, $W_g \in R^{d/4}$, $b_g \in R$. Thus, it can compute the general importance of memory cell itself and the relation between its neighbors in contrast to IM-LEMN.

3.1.4 Spatial-Temporal LEMN

In episodic tasks, the importance of memory also changes over time and tasks. Hence, we propose Spatio-Temporal LEMN (ST-LEMN) that uses a GRU over time to consider the historical importance as well as relative importance of each memory entry. We simply replace the hidden layer in Equation (11) with a GRU over time as follows:

$$h_{t,i} = GRU_{\theta_h}\left(f_{t,i}, h_{t-1,i}\right) \tag{13}$$

$$g_{t,i} = W_g h_{t,i} + b_g \tag{14}$$

$$\pi(m_i|M_t, x_t) = softmax\left(g_{t,i}\right) \tag{15}$$

3.1.5 Memory Update

Using aforementioned memory-retention mechanism, the agent samples the memory cell index i from the probability distribution $\pi(m_i|M_t, x_t)$. Then it erases the i-th memory cell and appends the input x_t as follows:

$$M_{t+1} = [m_1, \dots, m_{i-1}, m_{i+1}, \dots, m_N, x_t] \tag{16}$$

3.1.6 No Operation (NOP)

As done in (Gulcehre et al., 2016), for IM-LEMN we add a NOP memory cell at the end of memory to consider the case where the input is not written to the memory. For S-LEMN and ST-LEMN, we append x_t to the place of NOP such that it could be selected for removal.

We integrate these retention mechanisms into base memory-augmented neural networks to enable to efficiently maintain its external memory. The details of the base networks are given in the Experiment section. We use Asynchronous Advantage Actor-Critic (A3C, Mnih et al., 2016) with Generalized Advantage Estimation (GAE, Schulman et al., 2015) to train all models.

Chapter 4

Experiment

We evaluate the proposed retention agent on three different tasks in following subsections.

4.1 Maze

(Oh et al., 2016) proposed a task for memory-based deep reinforcement learning (RL) agents, where the agent should navigate through a 3D maze and enter the correct exit, and demonstrated that the agents with external memory outperform the agents without external memory. However, they assumed that the agent has a sufficiently large memory to store every observed cell, although the actual amount of information was not as much as the size of the memory. In this experiment, we compare the navigator agent with and without ST-LEMN. We followed the same experimental setup as used in (Oh et al., 2016) except that we use A3C algorithm (Mnih et al., 2016) with GAE (Schulman et al., 2015) instead of Q-Learning algorithm (Mnih et al., 2015) to train the agent, and used only three actions - Move forward, Look Left, Look Right - for expedited training. We use MQN (Memory Q-Network) and FRMQN (Feedback Recurrent Memory Q-Network) as base networks to see the effect of the model without recurrent connections and with recurrent connections. We compared the performance of these base models with learned retention and FIFO scheduling under two different environments.

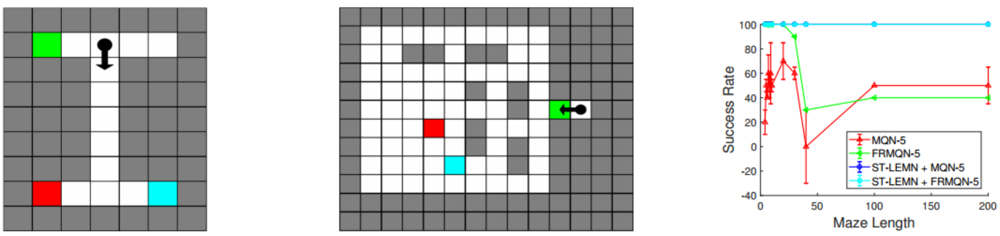


Fig 2. Illustration of Maze and result: *Example of IMaze environment (Left), Example of Random Maze environment (Center), Success rate of memory-based agents using FIFO memory scheduling and our retention mechanism (Right).*

4.1.1 IMaze

This environment contains an I-shaped maze, where the agent should reach the correct exit based on the color of the initial cell (Figure 2, Left). We train the models on mazes with varying lengths of corridors $N = \{5, 7, 9\}$, and validate on mazes with corridor with 15 different length $N = \{4, 6, 8, 10, 15, 20, 25, 30, 35, 40, 100, 200\}$. We set the memory size to 4 for all compared models. We observe that the agent with ST-LEMN successfully retains the indicator information in its memory while passing through the long corridor by removing useless frames that describe the corridor (Figure 3, First and 3, Second). At the end of corridor, agent with ST-LEMN retrieves stored indicator frame to decide which way to go. Agents with FIFO scheduling don't retain the indicator information and always exits at the same goal. FRMQN, which is an agent that has recurrent connection between time in its context embedding can complete task correctly on mazes with short corridor even with FIFO scheduling. Yet, it fails on mazes with long corridors since it is difficult to learn long-term dependencies without explicitly storing the cell in the memory. In contrast, we observed that the agents with ST-LEMN can solve the maze with small fixed-sized memories regardless of the length of maze, by learning the long-term importance of input data instances (Figure 2, Right).

4.1.2 Random Maze: Single Goal with Indicator

We also test with the randomly generated maze as in (Oh et al., 2016) (Figure 2, Center), testing for the Single Goal with Indicator (SingInd) task, where the agent should reach the correct goal based on the indicator that can be observed at the starting position. As shown in Table 1 the retention agent has significantly higher success rate compared to the agent with FIFO scheduling, as it retains the indicator frame while it navigates through the random maze. As shown in Figure 3 Frist and Second, the retention agent retains the indicator discovered at the start of maze in its memory and retrieves its information when it needs to make a decision.

Task	Test	Large
FIFO	68.90	60.40
ST-LEMN	**73.20**	**74.80**

Task	Test	Large
FIFO	84.30	91.20
ST-LEMN	**88.50**	**93.70**

Table 1. Best success rates on SingleInd Task: *MQN-5 (Left), FRMQN-5 (Right)*

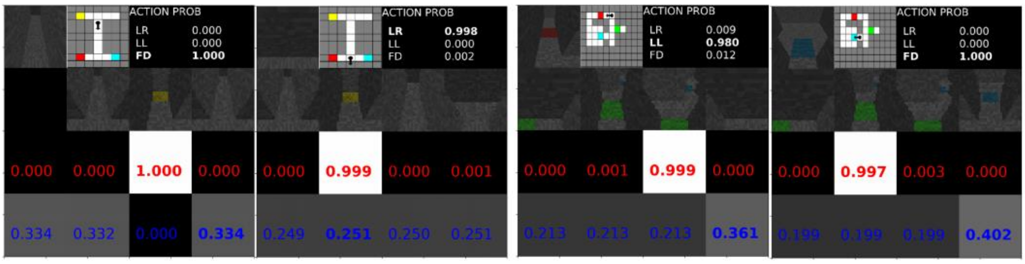


Fig 3. Illustration of visualization of MQN retention agent's status: *In the corridor (Frist), At the bottom fork (Second), In the middle (Third), In front of the goal (Fourth) (Frist-Second). I-Maze and (Third-Fourth) Single Goal with Indicator. The first row shows a current view of an agent, a top-down view of the maze with the current position of the agent, and action probability distribution. Each column from the second row shows the content of memory cells. The third and the fourth row indicate the attention value and the drop probability of each memory cell.*

4.2 Synthetic Episodic Question Answering

4.2.1 Dataset and Baseline Network

(Weston et al., 2015) created a synthetic dataset for episodic question answering, called bAbI, that consists of 20 tasks. Among tasks, we evaluate memory-retention mechanisms on two supporting facts task (task 2, Figure 4, Left). Additionally, We generate noisy and large version of the two supporting facts task from open-sourced template (Weston et al., 2015). Each task equally has five questions in an episode, where all questions share context sentences given in the episode. In the noisy two supporting fact task, which we refer to as **Noisy** task (Figure 4, Center), we inject noise sentences into original two supporting facts task to investigate the ability of retention mechanism to filter out the noise. We organize this synthetic dataset as follows: 60% of dataset have no noise sentence; 10% of dataset have approximately 30% noise sentences; 10% of dataset have approximately 45% noise sentences; 10% of dataset have approximately 60% noise sentences. Totally, the length of each episode is fixed to 45 (5 questions + 40 facts). Questions are placed after every 8 facts. In large and noisy two supporting facts task, which is called **Large** task (Figure 4, Right), composition of tasks is similar to Noisy but the length of episode and the position of questions vary. The length of each episode is randomly chosen between 20 and 80. Questions can be placed anywhere in episode.

idx	Memory
1	Mary journeyed to the bathroom
2	Sandra went to the garden
...	...
6	Sandra put down the milk there
Where is the milk?	garden [6, 2]
8	Daniel went to the garden
...	...
17	Daniel dropped the football
Where is the football?	bedroom [17, 12]
19	Sandra left the milk there
20	Daniel grabbed the football there
Where is the milk?	bedroom [19, 16]
22	Sandra grabbed the milk there
23	Daniel went to the kitchen
Where is the football?	kitchen [20, 23]
25	John travelled to the kitchen
26	Mary moved to the hallway
Where is the football?	kitchen [20, 23]

idx	Memory
1	Sandra moved to the kitchen
2	Wolves are afraid of cats
3	Sandra put down the apple
4	Sandra took the milk
5	Sandra is yellow
6	Mary is green
7	Sandra grabbed the apple
8	Mary went to the hallway
Where is the milk?	kitchen [1, 4]
...	...
40	Sandra let go of the football
41	Mary is green
42	Mary journeyed to the kitchen
43	Mary grabbed the milk
44	John is bored
Where is the apple?	kitchen [34, 42]

idx	Memory
1	Sandra is thirsty
2	Sandra is green
3	Sandra took the apple
4	John is bored
...	...
46	Mary is yellow
Where is the football?	kitchen [21, 31]
Where is the milk?	bedroom [36, 38]
49	Sandra is thirsty
50	John put down the football
...	...
74	Sandra is green
75	Sandra is a mouse
76	Sandra grabbed the apple
77	Cats are afraid of sheep
Where is the milk?	bedroom [36, 38]

Fig 4. Illustration of synthetic bAbI dataset: *Original task (Left), Noisy task (Center), Large task (Right). Sentences in green are noise sentences and ones in blue are supporting facts of each question.*

We use MemN2N (Sukhbaatar et al., 2015) as base networks. Particularly, we use base MemN2N with position encoding representation, 3 hops and adjacent weight tying. We set the dimension of memory representations to $d = 20$. We compare FIFO, IM-LEMN, S-LEMN and ST-LEMN on the two supporting facts task, and two modified tasks. IM-LEMN on this task uses an average of attention logits from each hop as z_t in Section 3.1.2. Also, it embeds input using GRU similar to a GRU controller in (Gulcehre et al., 2016). S-LEMN and ST-LEMN use the value memory of the first hop in the base Mem N2N as a memory representation $e_{t,i}$ in Section 3.1.3 and 3.1.4. We constrain the size of memory as 5 or 10 to validate the scheduling performance of our retention mechanism. Since we jointly train the agent for QA task and memory retention, for stable training we pretrain MemN2N with FIFO mechanism for 50k steps and then go on with training other mechanisms. We train our models using ADAM optimizer (Kingma & Ba, 2014) with the learning rate of 0.001 for 200k steps on the Original and Noisy tasks, and for 400k steps on the Large task.

4.2.2 Result

Table 2 shows the results on synthetic episodic question answering tasks. (Sukhbaatar et al., 2015) reports the lowest error by multiple agents to cope with large variance in performance, and we follow this evaluation measure. For the Original and Noisy tasks, we measure error rate using best performing parameter among three takes of training, and among five takes for the Large task. As shown in table 2, our suggested memory retention mechanism significantly outperforms naive policy and policy inspired by LRU in all cases. For more detailed analysis, we present two sampled data from IM-LEMN and ST-LEMN in Figure 5. IM-LEMN's selection policy seems to largely depend on the current context, and therefore it deletes not only noise sentences but also informative sentences as well. Compared to IM-LEMN, memory of ST-LEMN doesn't have any noise sentences in its memory but evenly contains informative facts about location and object acquisition.

Task	Original	Noisy
FIFO	41.40	50.20
IM-LEMN	23.70	40.20
S-LEMN	20.30	40.60
ST-LEMN	**17.50**	**11.70**

Task	Original	Noisy	Large
FIFO	16.50	44.10	32.40
IM-LEMN	16.10	18.90	9.00
S-LEMN	5.00	4.80	**5.10**
ST-LEMN	**4.60**	**3.90**	5.60

Table 2. Best error rates on bAbI tasks: *Memory size 5 (Left), Memory size 10 (Right)*

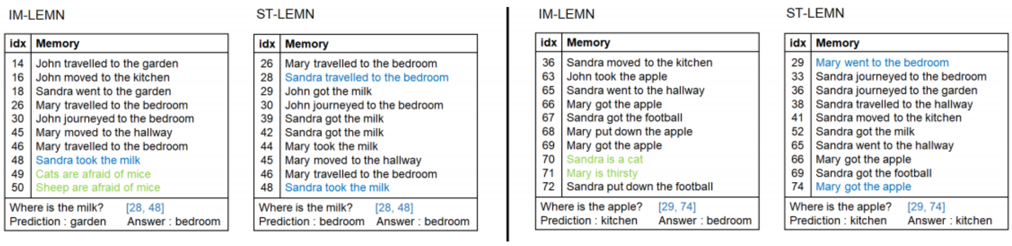


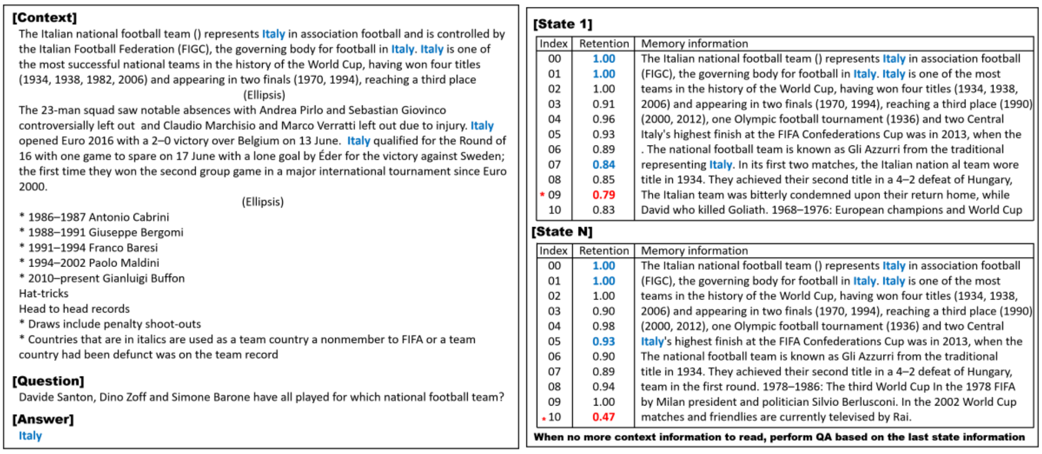
Fig 5. Illustration of example of memory state on the tasks: *Idx denotes index of sentence in one episode. Sentences in blue are supporting facts and ones in green are irrelevant ones.*

4.3 TriviaQA

4.3.1 Dataset and Baseline Network

TriviaQA (Joshi et al., 2017) is a question-answering dataset over 950K question-answer pairs in 662K evidence documents collected manually. The problem is difficult to solve by conventional models as it requires multiple reasoning with large number of sentences. The average number of words per document is 2895, which is infeasible to handle using existing reading comprehension models. We limit the number of words per document to 800, and used only questions that could be spanned within 400 words out of the 800 for training for expedited training. Also, we only use the first spanned word as labels since TriviaQA provides only the answer word and not their correct indices in the document. For a test set, we use all words in a document for each question-answer pair. Since the dataset doesn' provide labels for the test set, we use a validation set for test which contains both distant supervision set whose labels are collected automatically and verified set evaluated by the annotator. We evaluate our work only on the Wikipedia dataset, since previous work (Joshi et al., 2017), (Pan et al., 2017), (Yu et al., 2018) report similar results on both datasets.

As base network, we use BiDAF (Seo et al., 2016), which is one of the state-of-the-art reading comprehension model that predicts the indices for the exact location of the answer in the given document and modified the word-level intermediate representations to sentence-level representations using RNN to handle an entire sentence at a time. We set the memory size $N = 10$, and the word length per memory slot to 20. We trained our memory-augmented BiDAF using ADAM optimizer (Kingma & Ba, 2014) with the initial learning rate of 0.0001.



[Context]

The Italian national football team () represents Italy in association football and is controlled by the Italian Football Federation (FIGC), the governing body for football in Italy. Italy is one of the most successful national teams in the history of the World Cup, having won four titles (1934, 1938, 1982, 2006) and appearing in two finals (1970, 1994), reaching a third place

(Ellipsis)

The 23-man squad saw notable absences with Andrea Pirlo and Sebastian Giovinco controversially left out and Claudio Marchisio and Marco Verratti left out due to injury. Italy opened Euro 2016 with a 2–0 victory over Belgium on 13 June. Italy qualified for the Round of 16 with one game to spare on 17 June with a lone goal by Éder for the victory against Sweden; the first time they won the second group game in a major international tournament since Euro 2000.

(Ellipsis)

* 1986–1987 Antonio Cabrini
* 1988–1991 Giuseppe Bergomi
* 1991–1994 Franco Baresi
* 1994–2002 Paolo Maldini
* 2010–present Gianluigi Buffon

Hat-tricks

Head to head records

* Draws include penalty shoot-outs
* Countries that are in italics are used as a team country a nonmember to FIFA or a team country had been defunct was on the team record

[Question]

Davide Santon, Dino Zoff and Simone Barone have all played for which national football team?

[Answer]

Italy

[State 1]

Index	Retention	Memory information
00	1.00	The Italian national football team () represents Italy in association football
01	1.00	(FIGC), the governing body for football in Italy. Italy is one of the most
02	1.00	teams in the history of the World Cup, having won four titles (1934, 1938,
03	0.91	2006) and appearing in two finals (1970, 1994), reaching a third place (1990)
04	0.96	(2000, 2012), one Olympic football tournament (1936) and two Central
05	0.93	Italy's highest finish at the FIFA Confederations Cup was in 2013, when the
06	0.89	. The national football team is known as Gli Azzurri from the traditional
07	0.84	representing Italy. In its first two matches, the Italian nation al team wore
08	0.85	title in 1934. They achieved their second title in a 4–2 defeat of Hungary,
* 09	0.79	The Italian team was bitterly condemned upon their return home, while
10	0.83	David who killed Goliath. 1968–1976: European champions and World Cup

[State N]

Index	Retention	Memory information
00	1.00	The Italian national football team () represents Italy in association football
01	1.00	(FIGC), the governing body for football in Italy. Italy is one of the most
02	1.00	teams in the history of the World Cup, having won four titles (1934, 1938,
03	0.90	2006) and appearing in two finals (1970, 1994), reaching a third place (1990)
04	0.98	(2000, 2012), one Olympic football tournament (1936) and two Central
05	0.93	Italy's highest finish at the FIFA Confederations Cup was in 2013, when the
06	0.90	The national football team is known as Gli Azzurri from the traditional
07	0.89	title in 1934. They achieved their second title in a 4–2 defeat of Hungary,
08	0.94	team in the first round. 1978–1986: The third World Cup In the 1978 FIFA
09	1.00	by Milan president and politician Silvio Berlusconi. In the 2002 World Cup
* 10	0.47	matches and friendlies are currently televised by Rai.

When no more context information to read, perform QA based on the last state information

Fig 6. Illustration of TriviaQA dataset and the operation process of our model: *Question-answering pair and its context (Left), The operation process of our model (Right). During the process in Right, it retained the word 'Italy'(Thick blue) in order to answer a given question, by generating high retention value(Thick blue) on the memory cell containing 'Italy', and decides to erase uninformative memory cells with low retention value (Thick red / * Mark).*

4.3.2 Result

Table 3 shows the results on TriviaQA dataset. In (Joshi et al., 2017), they selected the highest score for ExactMatch and F1 score among multiple documents that could find the answer to a question. Our model outperforms all baselines because it has the ability to correctly identify informative sentences that are required to answer a given question. As shown in Figure 6, when new context information arrives at the model, our model determines which memory information is the most unnecessary based on the predicted the retention value.

Model	ExactMatch	F1score
FIFO	18.49	20.33
IM-LEMN	34.92	38.72
S-LEMN	42.97	46.61
ST-LEMN	**45.21**	**49.04**

Model	ExactMatch	F1score
FIFO	16.04	17.63
IM-LEMN	32.07	33.75
S-LEMN	38.36	41.27
ST-LEMN	**44.33**	**47.24**

Table 3. Q&A accuracy on TriviaQA: *Distant supervision (Left) and Verified (Right) subsets*

[Context]

Band Aid is a charity supergroup featuring mainly British and Irish musicians and recording artists. It was founded in 1984 by Bob Geldof and Midge Ure to raise money for anti-famine efforts in Ethiopia by releasing the song "Do They Know It's Christmas?" for the Christmas market that year. On 25 November 1984, the song was recorded at Sarm West Studios in Notting Hill, London, and was released in the UK four days later. The single surpassed the hopes of the producers to become the Christmas number one on that release. Two subsequent re-recordings of the song to raise further money for charity also topped the charts. The original was produced by Midge Ure. The 12" version was mixed by Trevor Horn. In November 2014, a new version of the song was recorded by artists under the name of Band Aid 30. Background

(Ellipsis)

Many of the people involved in the original Band Aid single appeared on the 1984 Christmas edition of Top of the Pops, along with Slade, The Thompson Twins, Bronski Beat, Frankie Goes to Hollywood and Jim Diamond to mime to the record. This was the only time the original Band Aid single was performed live on television. However Bono could not attend, which led to the spectacle of Paul Weller miming to Bono's line. According to the film made by The Tube on the days of the recording 24–25 November 1984, Bob Geldof says The Edge from U2 was to have played guitar on the track but was unable to as he was in hospital at the time with a kidney infection.

(Ellipsis)

'I'm not afraid to say that I think Band Aid was diabolical. Or to say that I think Bob Geldof is a nauseating character. Many people find that very unsettling, but I'll say it as loud as anyone wants me to. In the first instance the record itself was absolutely tuneless. One can have great concern for the people of Ethiopia, but it's another thing to inflict daily torture on the people of Great Britain. It was an awful record considering the mass of talent involved. And it wasn't done shyly it was the most self-righteous platform ever int he history of popular music.' View of Africa At the time of the 2004 release, the World Development Movement described the lyrics as "patronising, false and out of date". In 2014, musician, Fuse ODG, turned down a request to sing with the group during the 2014 ebola outbreak. He revealed that the lyrics of the song do not reflect what Africa truly is, citing lyrics such as "There is no peace and joy in west Africa this Christmas". According to him, he goes to Ghana yearly for the sole purpose of peace and joy, so singing such lyrics would be a blatant lie.

[Question]

In what year was Band-Aid's Do They Know It's Christmas the UK Christmas chart-topping record (bonus point each for the years of reissue success by Band Aid II and Band Aid 20)?

[Answer]

1984

[State 1]

Index	Retention	Memory information
00	1.00	Band Aid is a charity supergroup featuring mainly British and Irish musicians and recording artists. It was founded in
01	1.00	1984 by Bob Geldof and Midge Ure to raise money for anti-famine efforts in Ethiopia by releasing the song
02	0.91	"Do They Know It's Christmas?" for the Christmas market that year. On 25 November 1984
03	0.83	, the song was recorded at Sarm West Studios in No tting Hill, London, and was released in the UK
04	0.83	four days later. The single surpassed the hopes of the producers to become the Christmas number one on that
* 05	0.81	release. Two subsequent re-recordings of the song to raise further money for charity also topped the charts. The
06	0.98	original was produced by Midge Ure. The 12" version was mixed by Trevor Horn. In November 2014
07	0.97	, a new version of the song was recorded by artists under the name of Band Aid 30. Background
08	0.93	The supergroup was formed by Bob Geldof, former lead singer of Irish Band The Boomtown Rats. The BBC played
09	0.87	a major role in capturing the poverty affecting Ethiopian citizens and thereby influenced Geldof to take action. Paula Yates
10	0.86	, Bob Geldof's partner, is considered to have been the brains behind the original Band Aid.

[State 2]

Index	Retention	Memory information
00	1.00	Band Aid is a charity supergroup featuring mainly British and Irish musicians and recording artists. It was founded in
01	1.00	1984 by Bob Geldof and Midge Ure to raise money for anti-famine efforts in Ethiopia by releasing the song
02	0.87	"Do They Know It's Christmas?" for the Christmas market that year. On 25 November 1984
03	0.82	, the song was recorded at Sarm West Studios in No tting Hill, London, and was released in the UK
* 04	0.81	four days later. The single surpassed the hopes of the producers to become the Christmas number one on that
05	0.99	original was produced by Midge Ure. The 12" version was mixed by Trevor Horn. In November 2014
06	1.00	, a new version of the song was recorded by artists under the name of Band Aid 30. Background
07	0.97	The supergroup was formed by Bob Geldof, former lead singer of Irish Band The Boomtown Rats. The BBC played
08	0.85	a major role in capturing the poverty affecting Ethiopian citizens and thereby influenced Geldof to take action. Paula Yates
09	0.84	, Bob Geldof's partner, is considered to have been the brains behind the original Band Aid.
10	0.86	It was she who became the driving force that inspired (and helped) Geldof to rally the most famous

[State N-1]

Index	Retention	Memory information
00	1.00	Band Aid is a charity supergroup featuring mainly British and Irish musicians and recording artists. It was founded in
01	1.00	1984 by Bob Geldof and Midge Ure to raise money for anti-famine efforts in Ethiopia by releasing the song
02	0.91	"Do They Know It's Christmas?" for the Christmas market that year. On 25 November 1984
03	1.00	original was produced by Midge Ure. The 12" version was mixed by Trevor Horn. In November 2014
04	1.00	, a new version of the song was recorded by artists under the name of Band Aid 30. Background the
05	1.00	supergroup was formed by Bob Geldof, former lead singer of Irish Band The Boomtown Rats. The BBC played
06	0.85	a major role in capturing the poverty affecting Ethiopian citizens and thereby influenced Geldof to take action. Paula Yates
07	0.86	pop stars of the Eighties to raise money for famine relief in Ethiopia. The group was composed of forty
08	0.87	in a documentary, that he knew his opening lines were written for David Bowie, who was not
* 09	0.74	that the lyrics of the song do not reflect what Africa truly is, citing lyrics such as "There
10	0.77	is no peace and joy in west Africa this Christmas". According to him, he goes to Ghana

[State N]

Index	Retention	Memory information
00	1.00	Band Aid is a charity supergroup featuring mainly British and Irish musicians and recording artists. It was founded in
01	1.00	1984 by Bob Geldof and Midge Ure to raise money for anti-famine efforts in Ethiopia by releasing the song
02	0.92	"Do They Know It's Christmas?" for the Christmas market that year. On 25 November 1984
03	1.00	original was produced by Midge Ure. The 12" version was mixed by Trevor Horn. In November 2014
04	1.00	, a new version of the song was recorded by artists under the name of Band Aid 30. Background the
05	1.00	supergroup was formed by Bob Geldof, former lead singer of Irish Band The Boomtown Rats. The BBC played
06	0.86	a major role in capturing the poverty affecting Ethiopian citizens and thereby influenced Geldof to take action. Paula Yates
07	0.87	pop stars of the Eighties to raise money for famine relief in Ethiopia. The group was composed of forty
08	0.88	in a documentary, that he knew his opening lines were written for David Bowie, who was not
09	0.75	is no peace and joy in west Africa this Christmas". According to him, he goes to Ghana
* 10	0.73	yearly for the sole purpose of peace and joy, so singing such lyrics would be a blatant lie.

Fig 7. Illustration of the operation process on TriviaQA: *Our model should remain the answer words in the memory cells while reading the context*

Chapter 5

Conclusion

We considered the problem of learning from streaming data, where the size of the data is too large to fit into the memory of a memory-augmented network. To solve the problem of retaining important data instances, we proposed Long-term Episodic Memory Network (LEMN), which is able to remember data instances of long-term generic importance. Using reinforcement learning, LEMN learns to decide which memory entry to replace when the memory becomes full, based on both relative importance between memory entries and their historical importance. We validated our LEMN on three different tasks, namely path finding, episodic question answering and long question answering against rule-based memory scheduling methods as well as an RL-agent trained without consideration of relative and historic importance of memory entries, against which it significantly outperforms. Further analysis of LEMN shows that such good performance comes from its ability to retain instances of long-term importance. As future work, we plan to apply our model to dialogue generation task for conversational agents.

REFERENCES

[1] Chung, Junyoung, et al. "Empirical evaluation of gated recurrent neural networks on sequence modeling." arXiv preprint arXiv:1412.3555 (2014).

[2] Graves, Alex, Greg Wayne, and Ivo Danihelka. "Neural turing machines." arXiv preprint arXiv:1410.5401 (2014).

[3] Graves, Alex, et al. "Hybrid computing using a neural network with dynamic external memory." Nature 538.7626 (2016): 471.

[4] Gulcehre, Caglar, et al. "Dynamic neural turing machine with soft and hard addressing schemes." arXiv preprint arXiv:1607.00036 (2016).

[5] Hochreiter, Sepp, and Jürgen Schmidhuber. "Long short-term memory." Neural computation 9.8 (1997): 1735-1780.

[6] Joshi, Mandar, et al. "Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension." arXiv preprint arXiv:1705.03551 (2017).

[7] Kaiser, Łukasz, et al. "Learning to remember rare events." arXiv preprint arXiv:1703.03129 (2017).

[8] Kim, Youngjin, Minjung Kim, and Gunhee Kim. "Memorization Precedes Generation: Learning Unsupervised GANs with Memory Networks." arXiv preprint arXiv:1803.01500 (2018).

[9] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." arXiv preprint arXiv:1412.6980 (2014).

[10] Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." Nature 518.7540 (2015): 529.

[11] Mnih, Volodymyr, et al. "Asynchronous methods for deep reinforcement learning." International conference on machine learning. 2016.

[12] Oh, Junhyuk, et al. "Control of memory, active perception, and action in minecraft." arXiv preprint arXiv:1605.09128 (2016).

[13] Pan, Boyuan, et al. "MEMEN: multi-layer embedding with memory networks for machine comprehension." arXiv preprint arXiv:1707.09098 (2017).

[14] Rajpurkar, Pranav, et al. "Squad: 100,000+ questions for machine comprehension of text." arXiv preprint arXiv:1606.05250 (2016).

[15] Santoro, Adam, et al. "Meta-learning with memory-augmented neural networks." International conference on machine learning. 2016.

[16] Schulman, John, et al. "High-dimensional continuous control using generalized advantage estimation." arXiv preprint arXiv:1506.02438 (2015).

[17] Seo, Minjoon, et al. "Bidirectional attention flow for machine comprehension." arXiv preprint arXiv:1611.01603 (2016).

[18] Sukhbaatar, Sainbayar, Jason Weston, and Rob Fergus. "End-to-end memory networks." Advances in neural information processing systems. 2015.

[19] Vinyals, Oriol, et al. "Matching networks for one shot learning." Advances in Neural Information Processing Systems. 2016.

[20] Weston, Jason, et al. "Towards ai-complete question answering: A set of prerequisite toy tasks." arXiv preprint arXiv:1502.05698 (2015).

[21] Xiong, Caiming, Stephen Merity, and Richard Socher. "Dynamic memory networks for visual and textual question answering." International conference on machine learning. 2016.

[22] Yu, Adams Wei, et al. "QANet: Combining Local Convolution with Global Self-Attention for Reading Comprehension." arXiv preprint arXiv:1804.09541 (2018).